Homework #5 **COSC 260** **40 pts**

In handout 18 we studied the Software Development Process, including the activities in the Software Lifecycle. We looked at several examples and test cases that stepped us through the process. In this assignment, you will step through the design and implementation of the Encryption program. (The 2nd part of this homework is given at the end of Handout #19)

1. Begin with an implementation that mirrors the implementation in your handout. Much of this is done for you in the handout; some of the documentation may be shortened in the handout but you may not shorten it in your implementation. Your program shall contain:
    a. Method descriptions,
    b. A list of inputs and outputs, and
    c. Pseudocode.
2. Modify the Encryption program so that it uses the following encryption algorithm:
    a. Every letter (both uppercase and lowercase) converted to its successor except z and Z, which are converted to 'a' and 'A' respectively (i.e., a to b, b to c, …, y to z, z to a, A to B, B to C, …, Y to Z, Z to A)
    b. Every digit converted to its predecessor except 0, which is converted to 9 (i.e., 9 to 8, 8 to 7, … 1 to 0, 0 to 9)
    c. Everything else unchanged (not encrypted).
    d. Be sure both encryption and decryption (reverse it) work with this algorithm.
3. In the comments, list tests you used to test the program. Use equivalence classes that make sense for this algorithm, as well as edge cases between those equivalence tests.
4. I will test your program with an automated script. Your user interface must match the interface in the problem as shown in the example output below. *10 points will be deducted for programs that do not conform to this program requirement!*
5. Encryption.java takes in a text file and the user is prompted for the name of the file. The file shall not be changed by the program. The output of the encryption method shall be a text file named `encrypted.txt`. The output of the decryption method shall be a text file named `decrypted.txt`
6. Submit the single file: Encryption.java Do NOT zip it!
7. The expected console user interface is given below. After these two tests are run, the data contained in decrypted.txt shall match the text data contained in test.txt.

```
>> java Encryption
1. Encrypt a file
2. Decrypt a file
1
Enter file name: test.txt
>> java Encryption
1. Encrypt a file
2. Decrypt a file
2
Enter file name: encrypted.txt
>>
```